

CLAIMS

1. A system for remote management of manageable resources distributed across multiple application hosts comprising:
- an application manager in a first application host;
 - a master agent in a second application host; and,
 - a plurality of mini-agents in remote application hosts separate from said first and second application hosts;
- said master agent receiving from said application manager a management command to perform at least one management operation directed to at least one manageable resource;
- said master agent communicating said management command to a mini-agent in a remote application host containing said at least one manageable resource;
- said mini-agent commanding said at least one manageable resource to perform said at least one management operation responsive to receiving said management command from said master agent.
2. The system of claim 1, wherein said first application host is a Java Virtual Machine ("JVM").
3. The system of claim 1, wherein said remote application hosts include JVMs.
4. The system of claim 3, wherein said remote application hosts further include non-Java environments.
5. The system of claim 1, wherein said master agent comprises a Java Management Extension ("JMX") communications connector for communicating with said application manager and said mini-agents.

6. The system of claim 1, wherein said master agent comprises:
a JMX communications protocol adaptor for providing a protocol-adapted view of
said master agent to said application manager.

7. The system of claim 5, wherein said JMX communications connector comprises
a Java Remote Method Invocation communications interface.

8. The system of claim 1, wherein each manageable resource has a managed bean
("MBean") interface.

9. The system of claim 8, wherein each said mini-agent comprises an MBean
server exposing said MBeans to said master agent application through a JMX
communications connector.

10. A method of remotely managing manageable resources distributed across
multiple application hosts comprising:

in a master agent in a first application host, receiving from an application
manager in a second application host a management command for performing a
management operation directed to a manageable resource in a remote application host
having a mini-agent;

communicating said management command to said mini-agent in said remote
application host; and,

in said mini-agent, performing said management operation on said manageable
resource by accessing an exposed member of said manageable resource according to
said management operation.

11. The method of claim 10, wherein said step of communicating said management
command to said mini-agent comprises:

accessing said mini-agent through a JMX communications connector.

12. The method of claim 11, wherein said step of accessing said mini-agent comprises:

accessing said mini-agent through an RMI communications interface.

13. The method of claim 10, wherein said step of performing said management operation comprises:

accessing a member of the manageable resource through a management interface to said manageable resource according to said management operation.

14. The method of claim 13, wherein said management interface is a managed bean ("MBean") registered in an MBean server in said mini-agent.

15. The method of claim 14, wherein said step of accessing said method comprises: accessing a member of the manageable resource exposed to said mini-agent by said MBean through said MBean server according to said management operation.

16. A method for configuring a system for remote management of manageable resources distributed across multiple remote application hosts comprising:

starting a master agent in a first application host;

starting a plurality of mini-agents in a plurality of corresponding remote application hosts separate from said first application host, each corresponding remote application host containing one of said plurality of mini-agents;

communicatively linking each mini-agent to said master agent; and,

registering manageable resources in each remote application host with a corresponding mini-agent,

whereby the distributed manageable resources can be remotely managed by an

11 application manager in an application host remote from said remote application hosts
12 containing the manageable resources.

1 17. The method of claim 16, wherein said first application host is a Java Virtual
2 Machine ("JVM").

1 18. The method of claim 16, wherein the remote application hosts include JVMs.

1 19. The method of claim 18, wherein the remote application hosts further include
2 non-Java environments.

Sub A1
1 20. The method of claim 16, further comprising:
2 providing a managed bean ("MBean") interface to the manageable resources
3 through which selected members of the manageable resources can be accessed.

1 21. The method of claim 20, wherein said step of starting a plurality of mini-agents
2 comprises:
3 starting a managed bean ("MBean") server in each of said plurality of mini-
4 agents, said MBean server exposing MBeans in said corresponding remote application
5 hosts.

1 22. The method of claim 21, wherein said registering step comprises:
2 registering said MBeans with said MBean server.

1 23. The method of claim 16, wherein said step of communicatively linking each mini-
2 agent to said master agent comprises:
3 creating RMI connector servers in said master agent and each said mini-agent;
4 and,

5 creating RMI connector clients in said master agent and each said mini-agent;
6 said master agent communicating with said mini-agents and said mini-agents
7 communicating with said master agent through RMI interfaces created by said RMI
8 connector servers and said RMI connector clients.

1 24. A system for remote management of manageable resources distributed across
2 remote application hosts comprising:

3 a master agent in a first application host for receiving from an application
4 manager in a second application host management commands to perform management
5 operations directed to selected manageable resources; and,

6 a plurality of mini-agents in the remote application hosts, each remote application
7 host containing only one mini-agent;

8 said master agent communicating said received management commands to said
9 mini-agents in the remote application hosts;

10 said mini-agents communicating said received management commands to said
11 selected manageable resources.

1 25. The system of claim 24, wherein said first application host is a Java Virtual
2 Machine ("JVM").

1 26. The system of claim 24, wherein the remote application hosts comprise JVMs.

1 27. The system of claim 26, wherein the remote application hosts further comprise
2 non-Java environments.

1 28. The system of claim 24, wherein said master agent comprises a Java
2 Management Extension ("JMX") communications connector for communicating with
3 said mini-agents in the remote application hosts.

1 29. The system of claim 24, wherein said master agent comprises:

2 a JMX communications connector for providing a view of said master agent to a
3 JMX-compliant application manager; and,

4 a JMX communications protocol adaptor for providing a protocol-adapted view of
5 said master agent to an application manager residing in a non-JVM environment.

1 30. The system of claim 28, wherein said JMX communications connector comprises
2 a Java Remote Method Invocation ("RMI") communications interface.

1 31. The system of claim 24, wherein the manageable resources have a managed
bean ("MBean") interface.

1 32. The system of claim 31, wherein each mini-agent in a corresponding remote
2 application host comprises a managed bean server ("MBean Server") for exposing
3 MBeans contained in said corresponding remote application host to said master agent
4 through a JMX communications connector.

1 33. A machine readable storage, having stored thereon a computer program for
2 remotely managing manageable resources distributed across multiple application hosts,
3 said computer program having a plurality of code sections executable by a machine for
4 causing the machine to perform the steps of:

5 in a master agent in a first application host, receiving from an application
6 manager in a second application host a management command for performing a
7 management operation directed to a manageable resource in a remote application host
8 having a mini-agent;

9 communicating said management command to said mini-agent in said remote
10 application host; and,

11 in said mini-agent, performing said management operation on said manageable

resource by accessing an exposed member of said manageable resource according to said management operation.

34. The machine readable storage of claim 33, wherein said step of communicating said management command to said mini-agent comprises:

accessing said mini-agent through a JMX communications connector.

35. The machine readable storage of claim 34, wherein said step of accessing said mini-agent comprises:

accessing said mini-agent through an RMI communications interface.

36. The machine readable storage of claim 33, wherein said step of performing said management operation comprises:

accessing a member of said manageable resource through a management interface to said manageable resource according to said management operation.

37. The machine readable storage of claim 36, wherein said management interface is a managed bean ("MBean") registered in an MBean server in said mini-agent.

38. The machine readable storage of claim 37, wherein said step of accessing said method comprises:

accessing a member of the manageable resource exposed to said mini-agent by said MBean through said MBean server according to said management operation.

39. A machine readable storage, having stored thereon a computer program for configuring a system for remote management of manageable resources distributed across multiple remote application hosts, said computer program having a plurality of code sections executable by a machine for causing the machine to perform the steps

5 of:

6 starting a master agent in a first application host;
 7 starting a plurality of mini-agents in a plurality of corresponding remote
 8 application hosts separate from said first application host, each corresponding remote
 9 application host containing one of said plurality of mini-agents;
 10 communicatively linking each mini-agent to said master agent; and,
 11 registering manageable resources in each remote application host with a
 12 corresponding mini-agent,
 13 whereby the distributed manageable resources can be remotely managed by an
 14 application manager in an application host remote from said remote application hosts
 containing the manageable resources.

1 40. The machine readable storage of claim 39, wherein said first application host is a
 2 Java Virtual Machine ("JVM").

1 41. The machine readable storage of claim 39, wherein the remote application hosts
 2 include JVMs.

1 42. The machine readable storage of claim 41, wherein the remote application hosts
 2 further include non-Java environments.

1 43. The machine readable storage of claim 39, further comprising:
 2 providing a managed bean ("MBean") interface to the manageable resources
 3 through which selected members of the manageable resources can be accessed.

1 44. The machine readable storage of claim 43, wherein said step of starting a
 2 plurality of mini-agents comprises:
 3 starting a managed bean ("MBean") server in each of said plurality of mini-

agents, said MBean server exposing MBeans in said corresponding remote application hosts.

45. The machine readable storage of claim 44, wherein said registering step comprises:

registering said MBeans with said MBean server.

46. The machine readable storage of claim 39, wherein said step of communicatively linking each mini-agent to said master agent comprises:

creating RMI connector servers in said master agent and each said mini-agent;

and,

creating RMI connector clients in said master agent and each said mini-agent;

said master agent communicating with said mini-agents and said mini-agents communicating with said master agent through RMI interfaces created by said RMI connector servers and said RMI connector clients.